

## Manycore processors MALT

**MALT** (Manycore Architecture with Lightweight Threads) is a family of energy-efficient processors with hundreds of cores on a single chip. MALT processors family has models with scalar, vector and hybrid architectures. Solutions based on MALT come near to customized FPGA systems in terms of performance-per-watt, surpassing them by performance-per-dollar value. MALT architecture programming complexity is on a par with the universal x86/GPU/ARM computer systems and they show the performance-per-watt of a much higher order.

## Programming

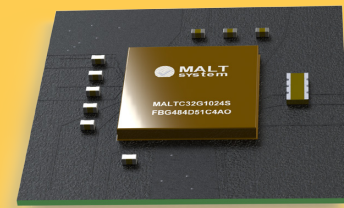
MALT programming is almost as easy as universal manycore processor programming. Depending on their preferences and requirements to application code optimization a programmer can choose one of the approaches:

- **C++ for MALT** (under development). It is the easiest way to start working with MALT. The C++17 standard is maintained to program scalar cores. Thread operations are implemented through the STL constructions (std:: thread, std:: mutex, etc.). MALT is considered a classic manycore processor, therefore it's easy to port the existing software and libraries.
- **OpenCL for MALT** (under development). OpenCL standard is being used for operations with scalar and vector MALT cores. The implementation is provided with a library of problem-oriented algorithms that are optimized for MALT. Now it is easier for AMD, NVIDIA, ARM users to switch to MALT.
- **MALTCC**. Functional analog of NVIDIA's NVCC. The personal front end for basic compilers of scalar and vector cores. MALTCC implements the potential of architecture and considerably simplifies program parallelization for target classes.

## Technical specification\*

- Performance up to 9.8 TFLOPS
- Operating frequency up to 1200 MHz
- Up to 256 universal cores
- Up to 1024 specialized cores
- Up to 8 MB of RAM on chip RAM
- Up to 96 GB of external DDR3 RAM
- Interfaces: PCIe, 1Gb Eth, SATA
- TDP up to 50 W
- 28 nanometer TSMC technology

\*Technical specification may depend on particular model.



## Application field

Solutions based on MALT may be performed as universal processors, nevertheless, they demonstrate the highest efficiency on the tasks they are designed for:

### Blockchain and cryptocurrency

**MALT-C** is a universal processor for complex cryptographic transformations, including blockchain transactions with utmost energy efficiency.

- Ethereum smart contract processing
- Delivery of trusted blockchain solutions
- Stream encryption, data integrity checking
- Modern cryptocurrency mining

### Big data

**MALT-D** are processors for simultaneous operations with a big data sets stored in random-access or external memory. MALT-D represents complex processing logic.

- Operations with extensive graph structures
- Massively parallel operation with B-trees
- Deep analysis of social networks
- Acceleration of SQL and NoSQL database operations

### Mathematical physics

**MALT-F** are processors for energy efficient solution of mathematical physics problems, requiring irregular memory access.

- Cellular automata for gas dynamic problems
- Adaptive grid computation
- Monte Carlo method in elementary particle physics
- Irregular sparse matrix problems



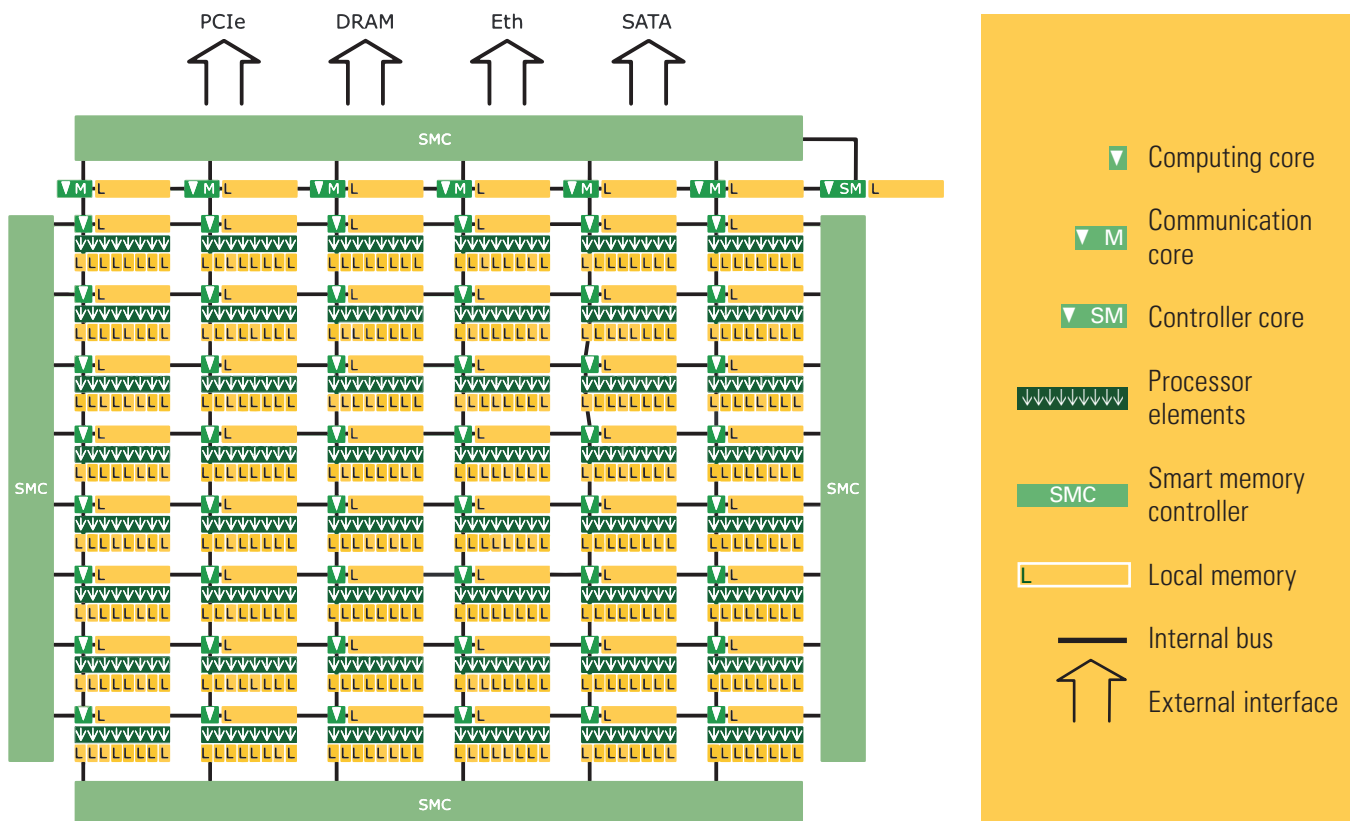
**MALT is a truly manycore processor**



## ARCHITECTURE

### How does it work?

Single-type computing elements produced ‘from scratch’ for a specific class of tasks are approximately 80% of any chip we develop. These elements determine productivity and chip power engineering. There are hundreds to several thousands cores on the chip! All the computing elements are programmed in C language or in its subset. The elements are combined into groups controlled by compact universal RISC processors grouped into computational array that to a programmer looks like a typical multi-threaded processor programmed in C/C++. The classic problems of multithreaded processing (conflict resolution mechanism of simultaneous access to shared data including atomic operations) are solved in hardware at the memory controller level.



### The Brief Look at MALT architecture

The basis of MALT architecture is formed by dozens to hundreds (depends on a model) compact asynchronous universal computing cores grouped by single or several original worm-hole networks with fat-tree topology. The communication between the networks is software and hardware. The universal core hierarchy consists of 3 levels: supermaster (controller core), master (communication cores), slave (computing cores available for user’s tasks. Slave cores may contain vector accelerators (depends on a model) performing specialized tasks of a target class. Each accelerator has 8 to 128 single-type computing elements with shared instruction memory. All computing cores and accelerators have own local data memory. All universal cores directly address DRAM and other shared resources (PCIe, Ethernet, SATA). The access arbitration to external resources is provided by smart memory controller (SMC) – a ‘smart memory’ hardware controller with an additional flag of data availability. The amount of SMC, the list and the configuration of shared external resources depend on MALT Family and are defined by requirements of the target tasks.



# MALT is a truly manycore processor